PROJECT 2

Daily Commute in Lisbon Metropolitan Area

TEAM LISBOA

18 municipalities

3 000 km2

2.8 M population

80%

OF THE PEOPLE LIVING
IN LISBON METROPOLITAN AREA
COMMUTE AT LEAST ONCE A DAY

Main Questions

What is the most efficient mode of transport to get to work?

Do longer commutes match longer distances?

Is the number of layovers related to the commute duration?

Main Challenges and Strenghts

Find Suitable Data

the available data is not always useful for the analysis we want to perform

Understand the data and clean it

assess the data and perform some data cleaning and manipulation before creating the database can be quite a challenge

Create database and perform analysis

create dataframes and perform calculations in search for answers for our questions

Data

National Statistics Institute

Mobility in Lisbon and Oporto Metropolitan Areas Survey (2017)

Issues

Inaccuracies of the data
pordata and lisboa aberta had no useful information

Limitations

no available data to analyze some questions we'd also like to answer

Database Structure

Average daily commute duration	Average daily commute distance	Average number of layovers per commute
• Mode of transport • Motive for commute • Average duration	• Mode of transport • Motive for commute • Average distance	• Mode of transport • Average number of layovers

Data Manipulation

Checking for NaN

```
# checking for NaN
work_duration.isnull().sum()
```

Rename columns

```
# rename column
duration_df = duration_df.rename(columns={"MUN":"Motive", "motor_bike":"motorbike","tax":"taxi"})
```

Drop columns

```
work_duration = work_duration.drop(columns=['other','taxi','plane'])
work_duration
```

Data Calculations

Calculating minimum, maximum and mean values

```python
# calculating avg commute duration for work
mean_duration = work_duration_t["Work"].mean()
mean_duration
```

Transposing dataframes

```python
work_distance_t = work_distance.transpose()
work_distance_t
```

Calculating relative values

```python
layover_df["0"]=layover_df["0"]*100/layover_df["Total"]
layover_df["1"]=layover_df["1"]*100/layover_df["Total"]
layover_df["> 2"]=layover_df["> 2"]*100/layover_df["Total"]
layover_df["Total"]=layover_df["Total"]*100/layover_df["Total"]
layover_df.head(10)
```

General Insights

Average daily commute distance	Average daily max commute distance	Average daily commute min distance
15 km	23 km	8 km
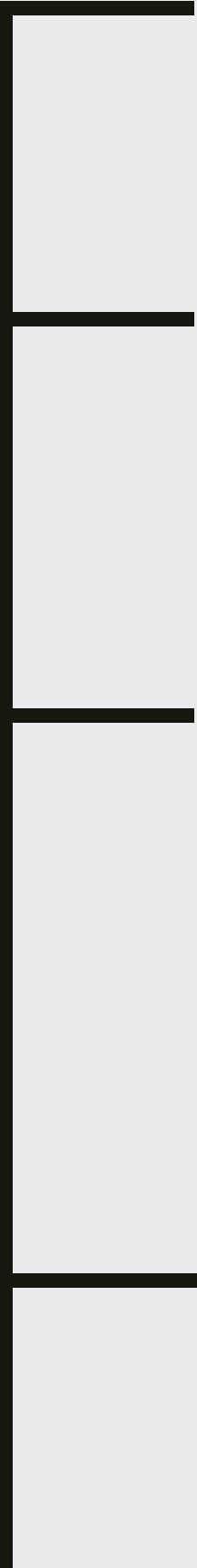
Average daily commute duration	Average daily max commute duration	Average daily commute min duration
35 min	54 min	20 min

Main Insights

Longer commutes = longer distances

More layovers = longer commutes
Less layovers = shorter commutes

Public transportation: boat is the slowest mode of transport and passengers have to 2 or more layovers per commute

Bike is the mode of transport with shortest times of commit and it's mostly used for shorter distances

Layover and Commute Duration

	transport	Total	0	1	> 2
0	car_driver	100.0	96.220762	3.565533	0.213705
1	car_pass	100.0	96.175160	3.424541	0.400301
2	motorbike	100.0	98.747164	1.221630	0.031206
3	taxi	100.0	94.813824	4.987164	0.199064
4	public_bus	100.0	55.334864	26.249279	18.415857
5	private_bus	100.0	87.693652	8.213424	4.092944
6	Train	100.0	17.679989	31.053200	51.266805
7	metro	100.0	49.298234	36.107084	14.594676
8	boat	100.0	8.600031	18.128642	73.271267
9	bike	100.0	92.641544	7.159921	0.198535

```python
# find the max layovers row in the "> 2" column
max_layovers = layover_df["> 2"].max()

# Find the mode of transportation with max layovers
max_layovers_row = layover_df.loc[layover_df['> 2'] == max_layovers].index.item()


## Then, given the index, we find the transport of the corresponding index
max_layovers_transport = layover_df.loc[max_layovers_row,'transport']

max_layovers_transport = max_layovers_transport.lower()
```

```python
min_layovers = layover_df["0"].max()

# Find the mode of transportation with max layovers
min_layovers_row = layover_df.loc[layover_df['0'] == min_layovers].index.item()

## Then, given the index, we find the transport of the corresponding index
min_layovers_transport = layover_df.loc[min_layovers_row,'transport']

min_layovers_transport = min_layovers_transport.lower()
```

Layover and Commute Duration

	Work	Back home
car_driver	24.8	22.1
car_pass	25.5	21.1
motorbike	19.5	18.4
public_bus	48.3	47.3
private_bus	33.2	37.2
train	51	54.5
metro	41.4	43.3
boat	54.2	60.9
bike	32.2	39.6

```python
max_duration = work_duration_t["Work"].max()

# Find the mode of transportation with max duration
max_duration_row = work_duration_t.loc[work_duration_t['Work'] ==
max_duration].index.item()

#max_duration_transport = max_duration_transport.lower()
```

```python
min_duration = work_duration_t_no_walk["Work"].min()

min_duration_row = work_duration_t_no_walk.loc[work_duration_t_no_
walk['Work'] == min_duration].index.item()
```

Layover and Commute Duration

Does a relation between duration of commute and number of layovers exists?

Answering to the question we made previously, we defined a if statment that prints a message.

```python
if max_layovers_transport == max_duration_row and min_layovers_transport == min_duration_row:
    print (f'The number of layovers influences the commute duration.')

else:
    print (f"there is no relationship between the number of layovers and commute duration")
```

```
The number of layovers influences the commute duration.
```

Layover and Commute Duration

Does a relation between duration of commute and number of layovers exists?

Answering to the question we made previously, we defined a if statment that prints a message.

```python
if max_layovers_transport == max_duration_row and min_layovers_transport == min_duration_row:
    print (f'The number of layovers influences the commute duration.')

else:
    print (f"there is no relationship between the number of layovers and commute duration")
```

```
The number of layovers influences the commute duration.
```

So.. which is the most efficient transport?

Transport	km	Time [min]	km/h	layovers (%)
Bike	8	32	15	92.64 | 0
Motorbike	13	20	36	98.75 | 0
Private Bus	23	33	42	87.69 | 0
Boat	19	54	34	73.27 | 2

Workflow

Research and data collection

Data cleaning and database creation

Data calculations

Insights and conclusions

THANK YOU

ANY QUESTIONS?